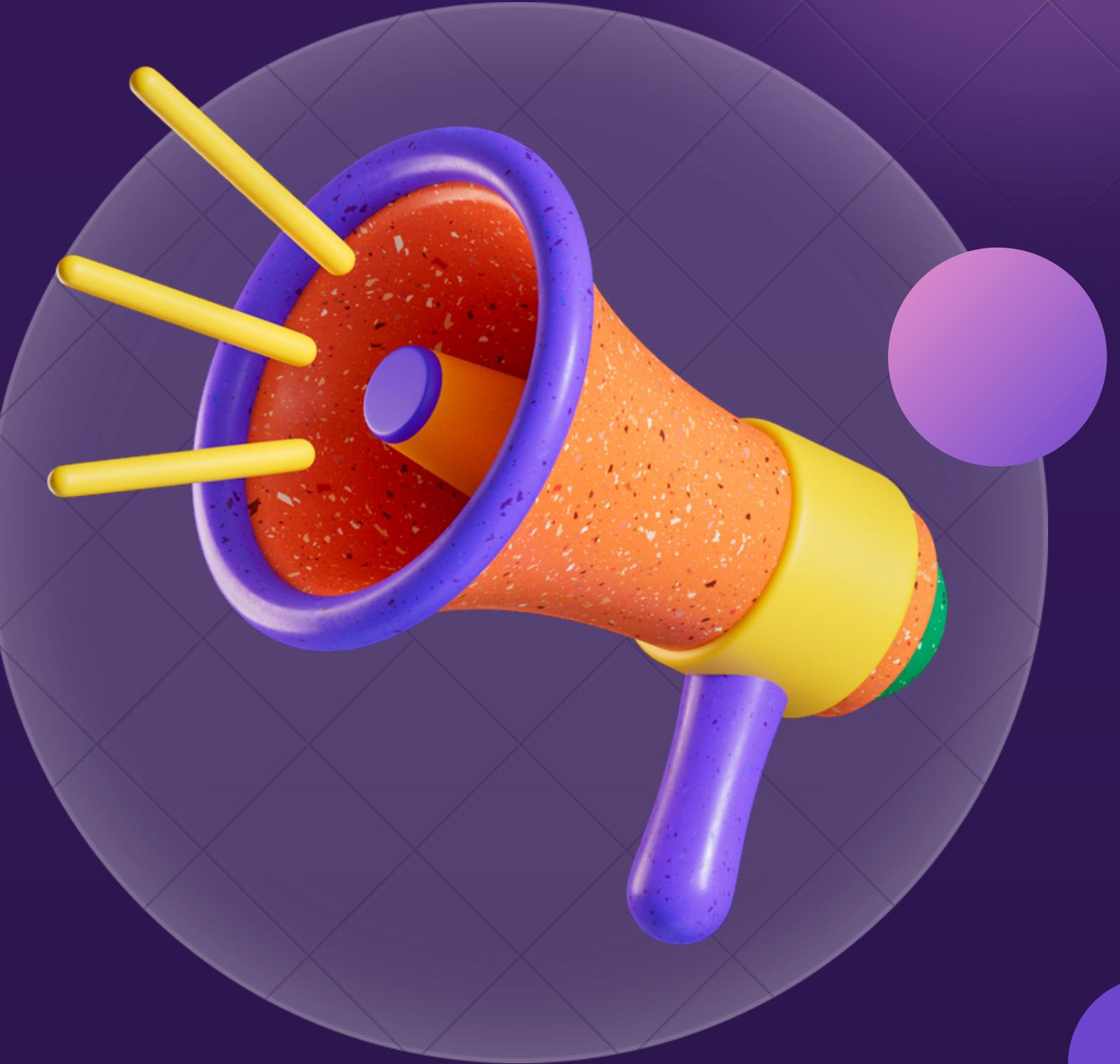




# Revolutionizing Event Management with a Comprehensive Web Application

Capstone Final Project



01

# Agenda

01 Introduction

02 Technologies

03 Requirements

04 IAAS, PaaS, SaaS

13 Performance & load  
Testing

05 Use of San Storage

06 Networking Infra

07 Fault Tolerance

08 Package & deploy

14 Monitoring Tool

09 Containerization

10 User & Admin Interface

11 Penetration Testing

12 Capacity Analysis

02



# About Our Project



Our project aims to develop a comprehensive web application for event management, providing users with a seamless platform to plan, organize, and execute various types of events. From corporate conferences to weddings and parties, our application will streamline the event planning process, making it easier and more efficient for both event organizers and attendees.



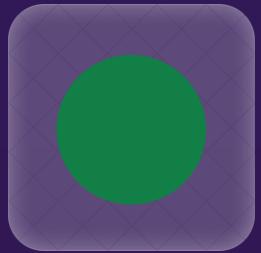
# Technologies We used

OpenStack: Cloud computing infrastructure.  
VMware: Virtualization software.  
Central Storage: Unified data storage solution.  
Docker: Containerization platform.  
Docker Swarm: Container orchestration tool.  
Front-end: HTML, CSS, JavaScript.  
Back-end: MySQL, Apache, PHP.  
Zabbix: Monitoring and testing tool.

[Next Page](#)

04

# IaaS, PaaS, SaaS



IAAS, PAAS and SAAS are 3 pillar which makes cloud computing better and efficient then traditional infrastructure.



IAAS & PASS

-- Shared Storage is the functionality which delivered this requirement.



PAAS

-- Our application hosted over the infrastructure is act as a PAAS for the end user.



## Overview

SAN Storages make it possible to store data in secure and independant manner then actual Serve

# Use of SAN Storage

The screenshot shows a web-based administrative interface for managing storage volumes. The top navigation bar includes 'Project' dropdown, 'Admin' dropdown, and a breadcrumb path 'Admin / Volume / Volumes'. The left sidebar has sections for 'Compute' (Overview, Volume, Volumes, Snapshots), 'Volume Types' (Volume Types, Groups, Group Snapshots, Group Types), and 'Groups'. The main content area is titled 'Volumes' and displays a table of three items. The table columns are: Project, Host, Name, Size, Status, Group, Type, Attached To, Bootable, Encrypted, and Actions. The data in the table is as follows:

Project	Host	Name	Size	Status	Group	Type	Attached To	Bootable	Encrypted	Actions
admin	block@lvm#LVM	ubuntu 3	15GiB	Available	-	_DEFAULT_		Yes	No	<button>Delete Volume</button>
admin	block@lvm#LVM	ubuntu 2	15GiB	Available	-	_DEFAULT_		Yes	No	<button>Delete Volume</button>
admin	block@lvm#LVM	ubuntu	15GiB	In-use	-	_DEFAULT_	/dev/vda on ubuntu	Yes	No	<button>Update Volume Status</button>

# Design Networking infrastructure

## IP Address table



Role	Machine Name	IP Address	Subnet Mask	Default Gateway
Controller Node	Controller	10.173.143.11	255.255.255.0	10.173.143.1
Compute Node	Compute	10.173.143.12	255.255.255.0	10.173.143.1
Block Storage	storage_block	10.173.143.22	255.255.255.0	10.173.143.1
Web server	webserver1	10.173.143.112	255.255.255.0	10.173.143.1
Database server	dbserver	10.173.143.113	255.255.255.0	10.173.143.1
Zabbix Monitoring server	monitorserver	10.173.77.16	255.255.255.0	10.173.77.1



# Fault Tolerance & Failover Capabilities



Docker start failing-demo				
TAINER ID	IMAGE	COMMAND	STATUS	NAMES
5effalfa	alpine	"sh -c 'sleep 10; ex..."	Up 6 seconds	failing-demo
5effalfa	alpine	"sh -c 'sleep 10; ex..."	Up 2 seconds	failing-demo
5effalfa	alpine	"sh -c 'sleep 10; ex..."	Up 7 seconds	failing-demo
5effalfa	alpine	"sh -c 'sleep 10; ex..."	Up 2 seconds	failing-demo
5effalfa	alpine	"sh -c 'sleep 10; ex..."	Up 6 seconds	failing-demo
5effalfa	alpine	"sh -c 'sleep 10; ex..."	Restarting (1) Less than a second	

01

## Docker Swarm

Container orchestration tool  
for managing Docker  
containers.

02

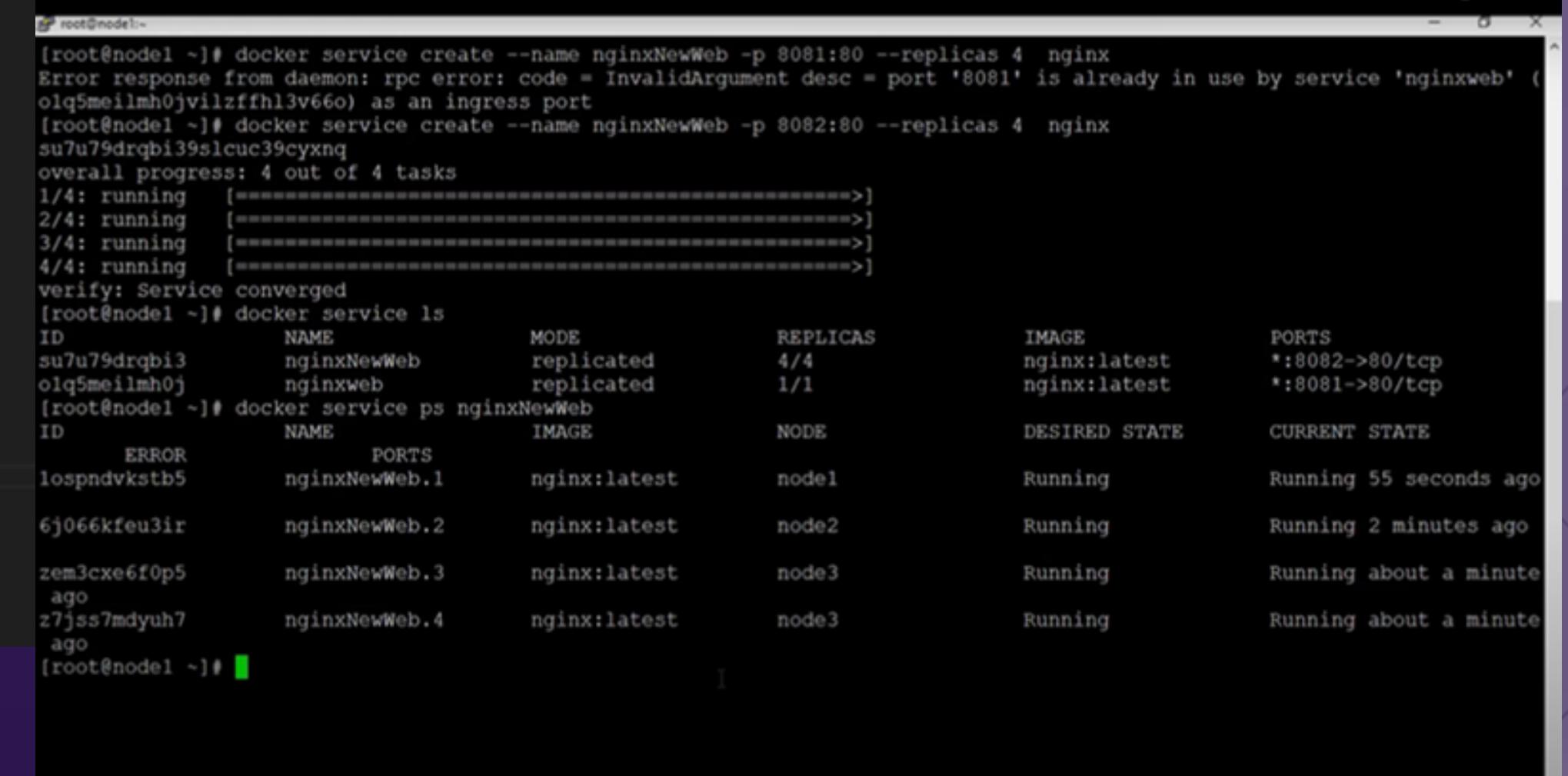
## Fault tolerance and failover capabilities

ensure system reliability by  
maintaining uninterrupted  
operation in the face of hardware  
or software failures

08



# Package and Deploy Finish Product

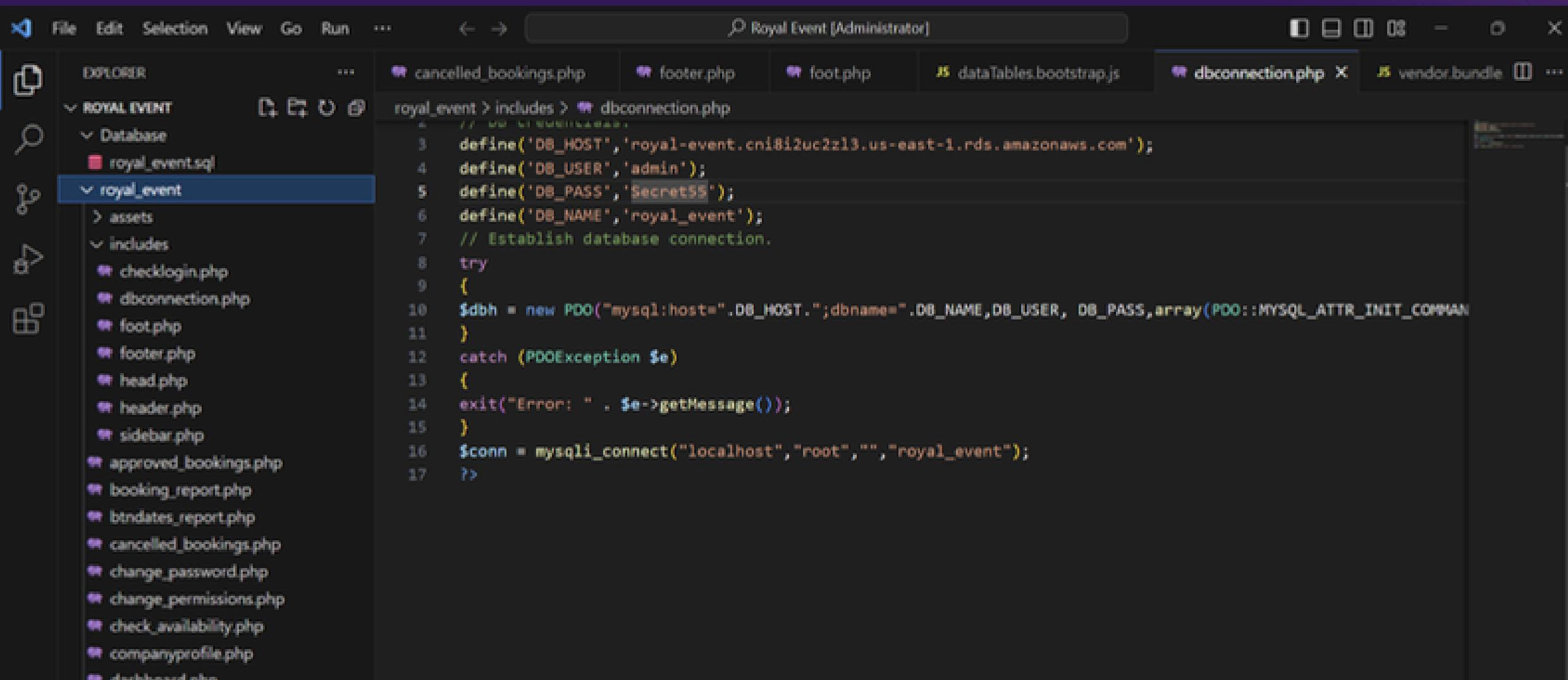


The terminal window shows the following commands and output:

```
[root@node1 ~]# docker service create --name nginxNewWeb -p 8081:80 --replicas 4 nginx
Error response from daemon: rpc error: code = InvalidArgument desc = port '8081' is already in use by service 'nginxweb' (olq5meilmh0jvilzffh13v66o) as an ingress port
[root@node1 ~]# docker service create --name nginxNewWeb -p 8082:80 --replicas 4 nginx
su7u79drqbi39slcuc39cyxng
overall progress: 4 out of 4 tasks
1/4: running [=====>]
2/4: running [======>]
3/4: running [======>]
4/4: running [======>]
verify: Service converged
[root@node1 ~]# docker service ls
ID          NAME      MODE      REPLICAS  IMAGE      PORTS
su7u79drqbi3    nginxNewWeb  replicated  4/4       nginx:latest *:8082->80/tcp
olq5meilmh0j    nginxweb   replicated  1/1       nginx:latest *:8081->80/tcp
[root@node1 ~]# docker service ps nginxNewWeb
ID          NAME      IMAGE      NODE      DESIRED STATE  CURRENT STATE
ERROR      PORTS
1ospndvkstb5  nginxNewWeb.1  nginx:latest  node1     Running        Running  55 seconds ago
6j066kfeu3ir  nginxNewWeb.2  nginx:latest  node2     Running        Running  2 minutes ago
zem3cxe6f0p5  nginxNewWeb.3  nginx:latest  node3     Running        Running  about a minute
ago
z7jss7mdyuh7  nginxNewWeb.4  nginx:latest  node3     Running        Running  about a minute
[root@node1 ~]#
```

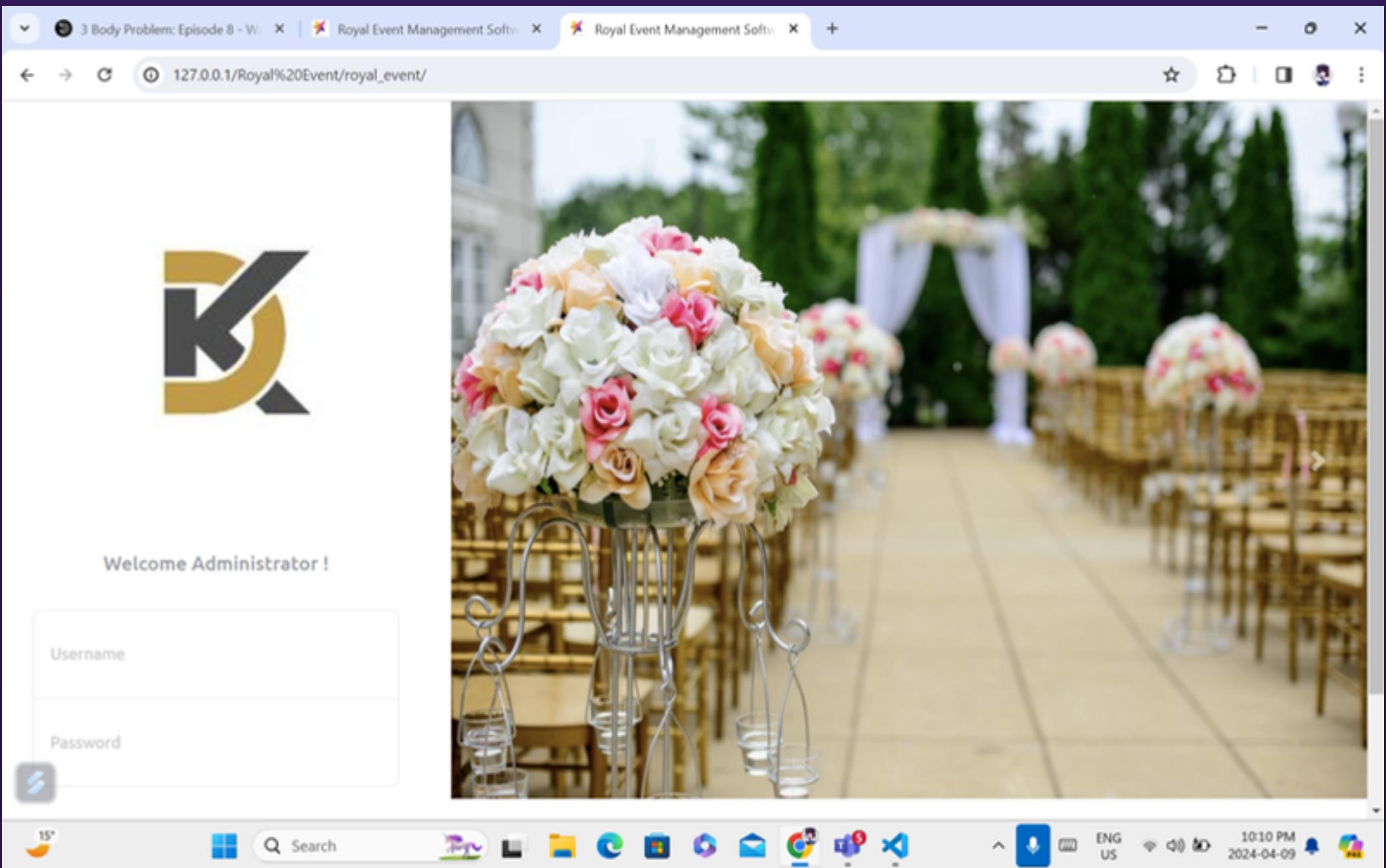


# Codes & Directory Structure

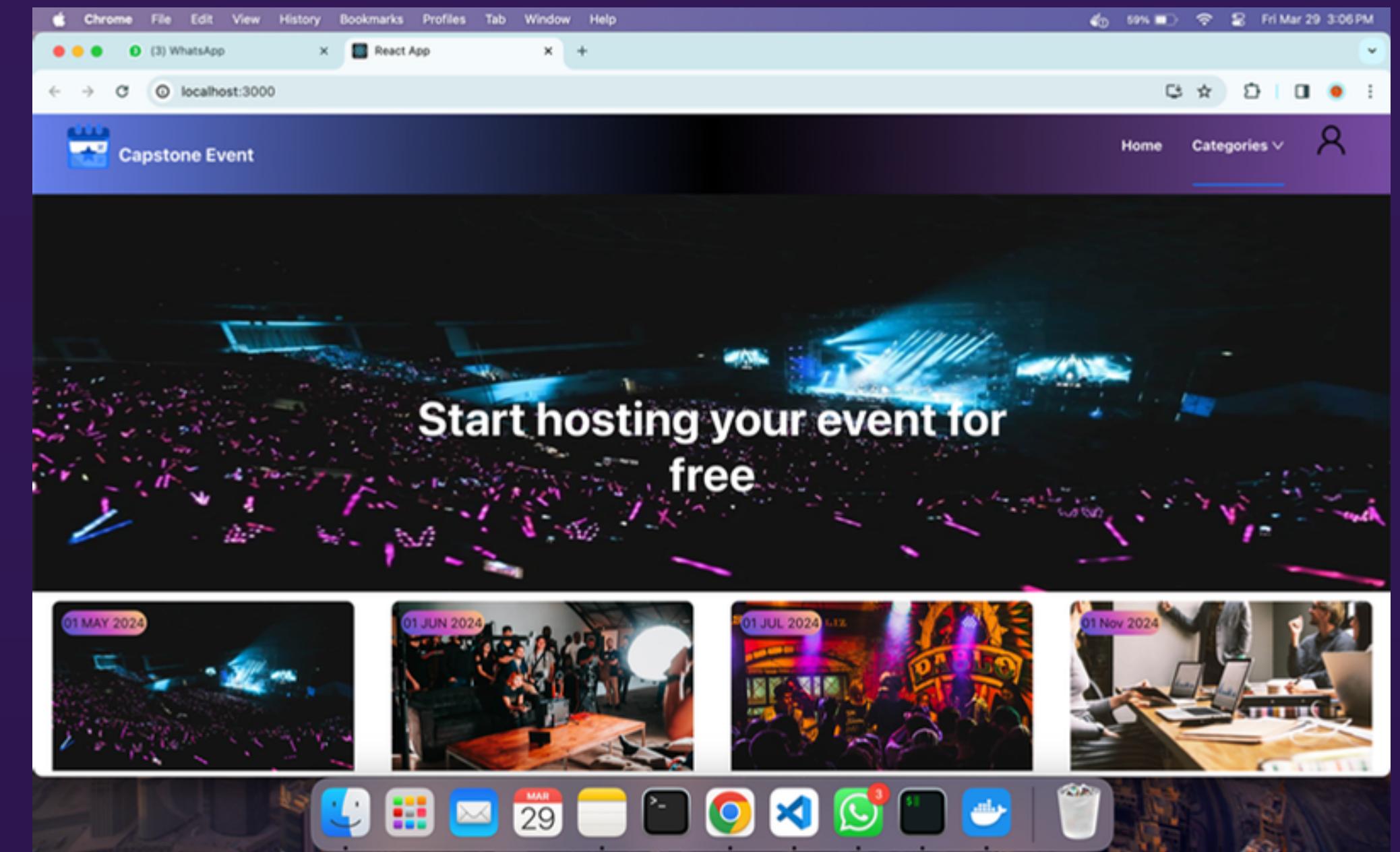


10

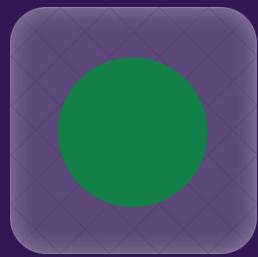
# Admin & User Interface



# Admin & User Interface



# Performance & load testing



Security Testing



Functional Testing



Performance Testing



Security Testing



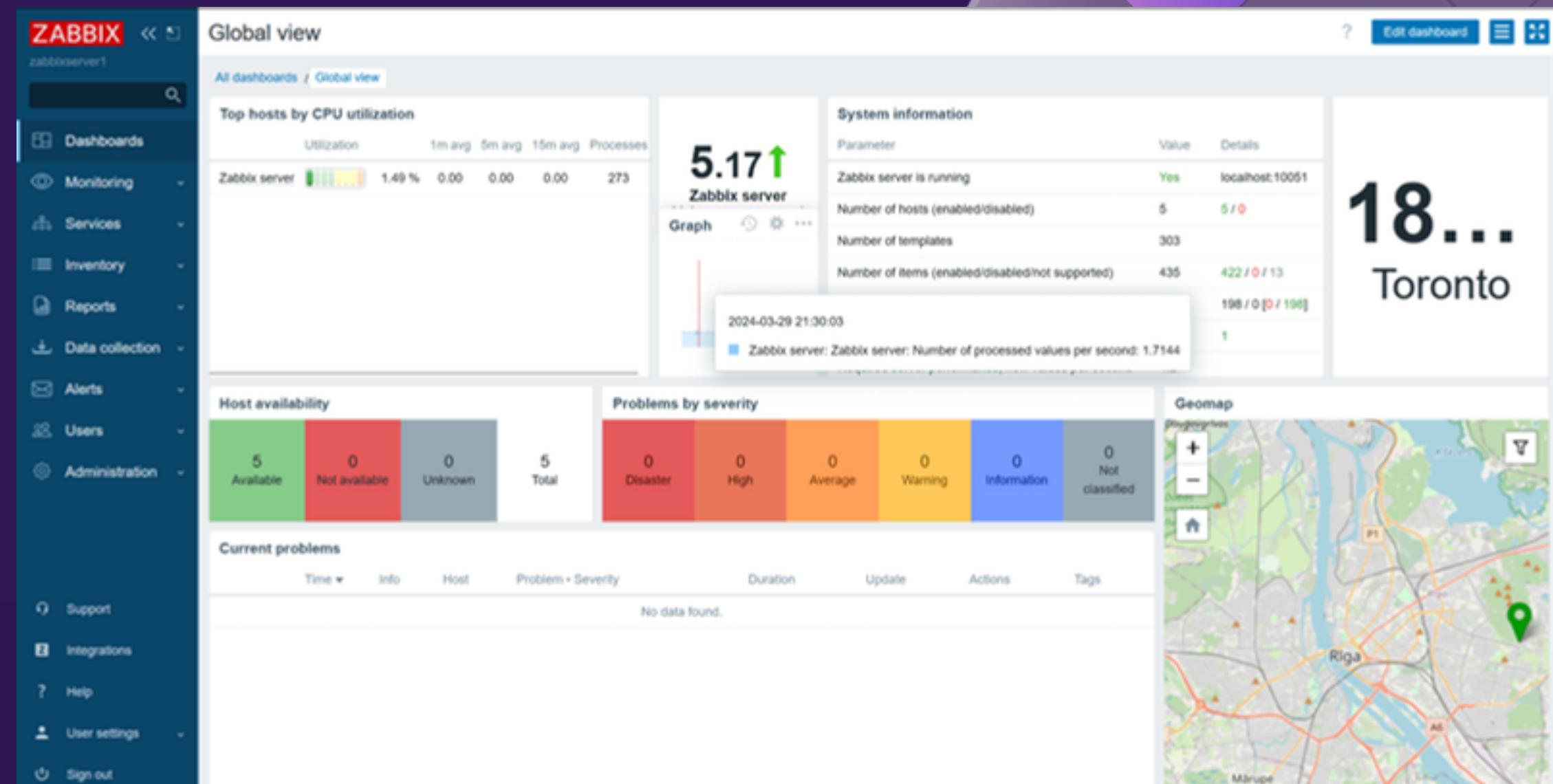
# Capacity Analysis and Planning

Capacity analysis and planning involve evaluating system performance data gathered during performance and load testing to determine if the current infrastructure can handle expected user loads and to plan for scaling resources accordingly to meet future demands.

[Next Page](#)

# Monitoring Tool

Comprehensive monitoring tool for tracking network, server, and application performance, providing real-time insights and alerts for proactive management.



# Thank You

Briefly elaborate on what you want  
to discuss.